UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/816.796	03/23/2001	Paul E. McKenney	BEA9-2001-0001-US1	5819

49056 7590 03/21/2007

LIEBERMAN & BRANDSDORFER, LLC
802 STILL CREEK LANE
GAITHERSBURG, MD 20878

EXAMINER	
MOLL, JESSE R	
ART UNIT	**PAPER NUMBER**
2181	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	03/21/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	Application No.	Applicant(s)	
	09/816,796	MCKENNEY, PAUL E.	
	Examiner	Art Unit	
	Jesse R. Moll	2181	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *20 December 2006*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-25* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-25* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 20 December 2006 has been entered.

Withdrawn Rejections

2. Applicant, via amendment, has overcome the rejection of claims 1-25 under 35 USC 112 first and second paragraphs. The rejections have been respectfully withdrawn.

3. Applicant, via amendment, has overcome the rejection of claims 12-25 under 35 USC 101. The rejection has been respectfully withdrawn.

Claim Rejections - 35 USC § 101

4. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 1-25 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

> Claims to computer-related inventions that are clearly nonstatutory fall into the same general categories as nonstatutory claims in other arts, namely natural phenomena such as magnetism, and abstract ideas or laws of nature which constitute "descriptive material. "Abstract ideas, Warmerdam, 33 F.3d at 1360, 31 USPQ2d at 1759, or the mere manipulation of abstract ideas, Schrader, 22 F.3d at 292-93, 30 USPQ2d at 1457-58, are not patentable. Descriptive material can be characterized as either "functional descriptive material" or "nonfunctional descriptive material." In this context, "functional descriptive material" consists of data structures and computer programs which impart functionality when employed as a computer component. (The definition of "data structure" is "a physical or logical relationship among data elements, designed to support specific data manipulation functions." The New IEEE Standard Dictionary of Electrical and Electronics Terms 308 (5th ed. 1993).) "Nonfunctional descriptive material" includes but is not limited to music, literary works and a compilation or mere arrangement of data (See MPEP section 2106, IV, B, i).

Claim 1 comprises steps of organizing, indicating, and forcing. They are just an abstract idea. The claim does not provide practical application that produces a useful, tangible and concrete result. Therefore, this claim is non-statutory. Similar problems exist in claims 2-11. Organizing does not explicitly require actual data movement but merely a reinterpretation of current data.

Examiner recommends replacing the limitation "forcing execution of said write operations to non-local memory to precede storing... to said new element of said shared resource in response to said indicating" in lines 8-10 to "executing said write operations to non-local memory prior to said storing... to said new element of said shared resource in response to said indicating" to add a tangible result to the claims. See below under the response to arguments section.

Claim Rejections - 35 USC § 102

5. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

6. Claims 1-25 are rejected under 35 U.S.C. 102(b) as being anticipated by Robertson (U.S. Patent No. 5,850,632), hereafter referred to as Robertson'632.

Referring to claims 1 and 12, Robertson'632 discloses as claimed a method for maximizing CPU performance in a multiprocessor (see Fig. 2), comprising organizing (such as placing the instructions in program order) data elements (instructions and other data) stored in a shared resource (any memory internal to the CPU; including the cache and latches [such as the program counter]) designed to support data manipulation functions (any non-read-only memory supports writing (which is a data manipulation function).

executing write operations in local memory (each of the instruction caches 21, 26, 31, and 36, see Fig. 2, is broadly interpreted as a local memory) to execute in an arbitrary order (see col. 16, lines 31-34, regarding the caches being fully associative. Note each of the instruction caches 21, 26, 31, and 36, see Fig. 2, is individually and locally used by the associated processors 71-74, see Fig. 2; and a full associative replacement algorithm is used to have an arbitrary order to replace the cache lines thereof) and at any time prior to storing a pointer (PC, program counter) from an existing

element of said shared resource (old instruction) to a new element of said shared resource (next instruction; stored by Program counter register in the Robertson'632's system) (note PC will be incremented or changed to a new number after the write operation; and when the program counter register saves a new PC value, it is interpreted as a new element) wherein said pointer is stored in said shared resource (in the program counter register); explicitly indicating a set of write operations to non-local memory (memory configuration cache 305, see Fig. 5, is broadly interpreted as a non-local memory since it is not disposed inside the processors 71-74, see Fig. 2) be conducted in a specified order (see col. 31, lines 38-57, regarding the specified order for writing (replacing) the cache lines of the memory configuration cache 305); and forcing executionsaid write operation to non-local memory (memory configuration cache 305, see Fig. 2, is broadly interpreted as a non-local memory since it is not disposed inside the processors 71-74, see Fig. 2) to precede storing said pointer from an existing element of a shared resource to said new element of said shared resource (note the forcing step occurs when the Robertson'632's system also has a memory access (write operations) to memory configuration cache 305) in response to said indicating (memory operations are executed in a specific order and executed in response to that order).

As to claims 2, 13, and 23, Robertson'632 also discloses: assigning first and second registers of a CPU for storing associated first and second instruction addresses (note the Robertson'632's processor certainly comprises registers such as PC (program counter), MAR (memory address register), General Data/Address Registers, or CAR (control address register for storing instruction addresses).

As to claim 3, Robertson'632 also discloses: providing a third instruction referencing said registers (this is the situation when the registers, such as PC (program counter), MAR (memory address register), General Data/Address Registers, or CAR (control address register) storing the first and second instruction addresses is referred to as the source or destination registers in a third instruction).

As to claims 4 and 14, Robertson'632 also discloses: said third instruction specifies ordering between said first and second instructions (this is the situation when the registers storing the first and second instruction addresses are referred to as the destination registers in a third LOAD instruction, therefore, certainly operating the ordering between said first and second instructions).

As to claims 5 and 15, Robertson'632 also discloses: said third instruction indicates said first instruction's execution attaining a first specified state of execution prior to said second instruction's execution attaining a second specified state of execution (note this occurs in the Robertson'632's system when either one of the first instruction and the second instruction depends from the other and each instruction's execution invokes its state of execution specified by such as its opcode).

As to claims 6 and 16, Robertson'632 also discloses: said first and said second specified states of execution are selected from the group consisting of: committing instruction execution, initiating memory access, completing a memory access, initiating an I/0 access, completing an I/O access, and completing instruction execution (note this occurs in the Robertson'632's system when the first instruction and the second instruction is a memory load/store operation and it therefore certainly involves at least

one of initiating memory access, completing a memory access as claimed).

As to claims 7, 17, and 24, Robertson'632 also discloses: assigning a sequence number to an associated instruction for maintaining instruction ordering (note this is the situation in the Robertson'632's system when a sequence of program to be executed is saved in the Robertson'632's main memory and each instruction in the program is assigned by a logical address or physical address number)

As to claims 8 and 18, Robertson'632 also discloses: statically encoding said sequence number within said instruction (inherently existing in the Robertson'632's processor when a sequence of program is therein).

As to claims 9 and 19, Robertson'632 also discloses: dynamically encoding said sequence number within said instruction (as set forth above, inherently the Robertson'632's processor comprises registers such as MAR (memory address register), General Data/Address Registers, or CAR (control address register for storing instruction addresses and for dynamically encoding the sequence number).

As to claims 10 and 20, Robertson'632 also discloses: placing a range of instructions into a hierarchical ordering system (note the control unit of the Robertson'632's CPU is reasonably and broadly interpreted as a hierarchical ordering system and a range of instructions is inherently placed in a process table).

As to claims 11, 21 and 25, Robertson'632 also discloses: implementing a special instruction for maintaining a hierarchical execution of said instruction (such as a microinstruction, existing in the Robertson'632's processor for control the instruction execution, which is broadly interpreted as a special instruction for maintaining a

hierarchical execution).

Referring to claim 22, Robertson'632 discloses as claimed a processor for use in a multiprocessor computer system (see Fig. 2), comprising: a first instruction for allowing write operations in local memory (each of the instruction caches 21, 26, 31, and 36, see Fig. 2, is broadly interpreted as a local memory) to occur in an arbitrary order (see col. 16, lines 31-34, regarding the caches being fully associative. Note each of the instruction caches 21, 26, 31, and 36, see Fig. 2, is individually and locally used by the associated processors 71-74, see Fig. 2; and a full associative replacement algorithm is used to have an arbitrary order to replace the cache lines thereof), a second instruction for explicitly indicating a set of write operations to non-local memory (memory configuration cache 305, see Fig. 2, is broadly interpreted as a non-local memory since it is not inside the processors 71-74, see Fig. 2) to be conducted in a specified order (see col. 31, lines 38-57, regarding the specified order for writing (replacing) the cache lines of the memory configuration cache 305), wherein write operations to non-local memory must execute prior to storage of a pointer (PC, program counter) from an existing element of a shared resource (old instruction) to a new element of a shared resource (next instruction; stored by Program counter register in the Robertson'632's system) (note PC will be incremented or changed to a new number after the write operation); and a third instruction for managing order of execution of said first and second instructions (note the above limitations are disclosed by Robertson'632 as set forth above in claim 4); wherein execution of said second instruction is responsive to said first instruction reaching a specified state of execution and said

specified state of execution is selected from the group consisting of: committing instruction execution, initiating memory access, completing a memory access, initiating an I/O access, completing an 1/0 access, and completing instruction execution (note the above limitations are disclosed by Robertson'632 as set forth above in claims 5 and 6).

Response to Arguments

7. Applicant's arguments filed 20 November 2006 have been fully considered but they are not persuasive.

8. Regarding the arguments directed to the rejection of claims 1-11 under 35 USC 101, Examiner disagrees. As stated above, the newly added limitation of "organizing data elements" does not add a tangible and concrete result. Reinterpreting how data is organized is not in itself concrete. While executing said write operations would be a concrete result a method, forcing execution of said write operation is merely claiming the actions which cause the execution of said write operation. These actions are not limited to be concrete. For example, forcing execution could merely be deciding to execute, which is not a concrete and tangible result.

9. Regarding the augments directed to the storage location of said pointer, Examiner disagrees. The claim recites the limitation "wherein the pointer is stored in said shared resource". Applicant is correct in pointing out that the memory configuration cache of Robertson does not have a program counter. However, there exist no

limitations in claim 1 that prohibit "said shared resource" from including the program counter register.

10. Regarding the augments directed to limitation "forcing", Examiner disagrees. The definition of the word "force" according to Dictionary.com Unabridged (v 1.1) is "to bring about of necessity or as a necessary result". Using this definition, it is reasonable to consider any circuitry that determines the order of memory operations to be "forcing" the memory operation to execute in a certain order. Alternatively, it is reasonable to say that an ordering circuit would be "forcing' a memory operation to execute out of order because it is contrary to normal program flow. Claim 1 is reasonably broad enough to cover both interpretations.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Jesse R. Moll whose telephone number is (571)272-2703. The examiner can normally be reached on M-F 10:00 am - 6:30 pm EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Donald A. Sparks can be reached on (571)272-4201. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Jesse R Moll
Examiner
Art Unit 2181

JM 3/19/07

DONALD SPARKS
SUPERVISORY PATENT EXAMINER